Initial Algebras Unchained

A Novel Initial Algebra Construction Formalized in Agda


Thorsten Wissmann
thorsten-wissmann.de

Stefan Milius
stefan-milius.eu

FAU Erlangen-Nürnberg, Germany

November 2024

Initial Algebras Unchained

A Novel Initial Algebra Construction Formalized in Agda

Thorsten Wissmann
thorsten-wissmann.de

Stefan Milius
stefan-milius.eu

FAU Erlangen-Nürnberg, Germany

Initial Algebras Unchained

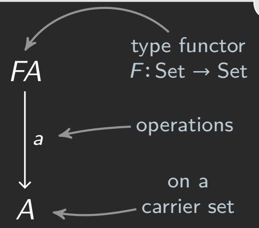
A Novel Initial Algebra Construction Formalized in Agda

Initial Algebras Unchained

A Novel Initial Algebra Construction Formalized in Agda

Initial Algebras Unchained

A Novel Initial Algebra Construction Formalized in Agda



E.g. $FX = \{\bullet\} + X \times X$

E.g. $A = \mathbb{N}$
$a(\bullet) = 0$
$a(\ell, r) = 1 + \max(\ell, r)$

Initial Algebras Unchained

A Novel Initial Algebra Construction Formalized in Agda

E.g. $FX = \{\bullet\} + X \times X$

T = all binary trees

constructors → $FT \xrightarrow{t} T$ ← data structures of type F

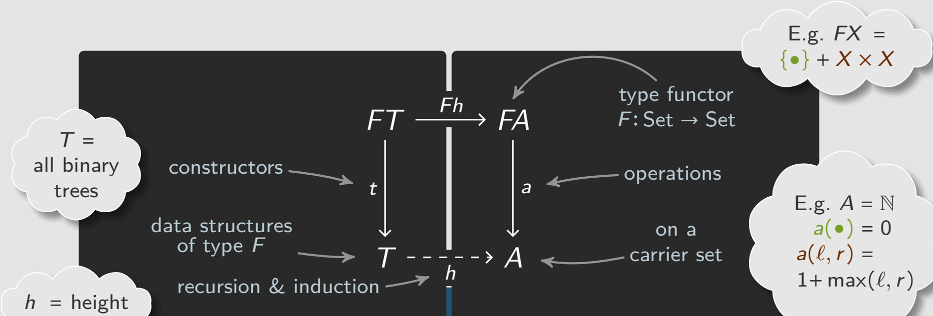
type functor $F\colon \mathsf{Set} \to \mathsf{Set}$ → FA

operations → $FA \xrightarrow{a} A$ ← on a carrier set

E.g. $A = \mathbb{N}$
$a(\bullet) = 0$
$a(\ell, r) = 1 + \max(\ell, r)$

Initial Algebras Unchained

A Novel Initial Algebra Construction Formalized in Agda



Initial Algebras Unchained

A Novel Initial Algebra Construction Formalized in Agda

E.g. $FX = \{\bullet\} + X \times X$

$$\begin{array}{ccc} FT & \xrightarrow{Fh} & FA \\ {\scriptstyle t}\downarrow & & \downarrow{\scriptstyle a} \\ T & \xrightarrow[h]{} & A \end{array}$$

type functor $F\colon \mathsf{Set} \to \mathsf{Set}$

T = all binary trees

constructors

operations

data structures of type F

on a carrier set

recursion & induction

h = height

E.g. $A = \mathbb{N}$

$a(\bullet) = 0$

$a(\ell, r) = 1 + \max(\ell, r)$

Initial Algebras Unchained

A Novel Initial Algebra Cons

Theorem (Adámek '74).

initial F-algebra

$=$

$\varnothing \to F\varnothing \to \dots F^{\omega}\varnothing \to \cdots$

by transfinite induction

☹ not in type-theory e.g. Agda

Initial Algebras Unchained

T = all binary trees

h = height

E.g. $FX = \{\bullet\} + X \times X$

E.g. $A = \mathbb{N}$
$a(\bullet) = 0$
$a(\ell, r) = 1 + \max(\ell, r)$

$$\begin{array}{ccc} FT & \xrightarrow{Fh} & FA \\ t \downarrow & & \downarrow a \\ T & \overset{h}{\dashrightarrow} & A \end{array}$$

type functor $F\colon \mathsf{Set} \to \mathsf{Set}$

constructors → t

operations → a

data structures of type F → T

on a carrier set → A

recursion & induction → h

Theorem (New)

initial F-algebra
=
colimit of all 'finite' recursive F-coalgebras

fully formalized in Agda

Theorem (Adámek '74).

initial F-algebra
=
$\emptyset \to F\emptyset \to \ldots F^{\omega}\emptyset \to \cdots$
by transfinite induction

not in type-theory e.g. Agda

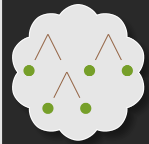
Theorem. For an accessible $F\colon \mathscr{C} \to \mathscr{C}$ on an accessible category $\mathscr{C}$ having binary coproducts of presentable objects,

initial F-algebra = colimit of all recursive F-coalgebras with presentable ('finite') carrier

Theorem. For an accessible $F\colon \mathscr{C} \to \mathscr{C}$ on an accessible category $\mathscr{C}$ having binary coproducts of presentable objects,

initial F-algebra = colimit of all recursive F-coalgebras with presentable ('finite') carrier

Recursive F-Coalgebra (R, r)



$$\begin{array}{ccc} FR & \xrightarrow{Fh} & FA \\ r \uparrow & & \downarrow \forall\, a \\ R & \xrightarrow{\exists! h} & A \end{array}$$

☺ well-founded induction

Osius '74

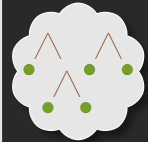
☺ divide & conquer algorithms (e.g. quicksort)

Capretta, Uustalu, Vene '06

Theorem. For an accessible $F\colon \mathscr{C} \to \mathscr{C}$ on an accessible category $\mathscr{C}$ having binary coproducts of presentable objects,

initial F-algebra = colimit of all recursive F-coalgebras with presentable ('finite') carrier

Recursive F-Coalgebra (R, r)



$$\begin{array}{ccc} FR & \xrightarrow{Fh} & FA \\ {\scriptstyle r}\uparrow & & \downarrow{\scriptstyle \forall\, a} \\ R & \overset{\exists! h}{- - - \to} & A \end{array}$$

☺ well-founded induction

Osius '74

☺ divide & conquer algorithms (e.g. quicksort)

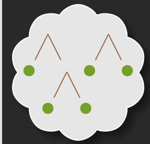
Capretta, Uustalu, Vene '06

Finiteness in a category $\mathscr{C}$

Theorem. For an accessible $F\colon \mathscr{C} \to \mathscr{C}$ on an accessible category $\mathscr{C}$ having binary coproducts of presentable objects,

initial F-algebra = colimit of all recursive F-coalgebras with presentable ('finite') carrier

Recursive F-Coalgebra (R, r)



$$\begin{array}{ccc} FR & \xrightarrow{Fh} & FA \\ r \uparrow & & \forall \downarrow a \\ R & \overset{\exists! h}{\dashrightarrow} & A \end{array}$$

☺ well-founded induction

Osius '74

☺ divide & conquer algorithms (e.g. quicksort)

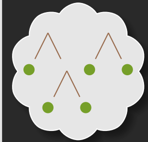
Capretta, Uustalu, Vene '06

Finiteness in a category $\mathscr{C}$

Theorem. For an accessible $F\colon \mathscr{C} \to \mathscr{C}$ on an accessible category $\mathscr{C}$ having binary coproducts of presentable objects,

initial F-algebra = colimit of all recursive F-coalgebras with presentable ('finite') carrier

Recursive F-Coalgebra (R, r)



$$\begin{array}{ccc} FR & \xrightarrow{Fh} & FA \\ r \uparrow & & \downarrow \forall a \\ R & \xrightarrow{\exists ! h} & A \end{array}$$

☺ well-founded induction

Osius '74

☺ divide & conquer algorithms (e.g. quicksort)

Capretta, Uustalu, Vene '06

Finiteness in a category $\mathscr{C}$

Fix a class Fil of filtered diagrams

~~require cardinals~~

X (Fil-)presentable :⟺ $\mathscr{C}(X, -)$ preserves colimits of diagrams in Fil

finite

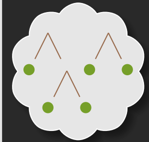
$\mathscr{C}$ (Fil-)accessible :⟺ build all objects from a set $\mathscr{C}_p$ of presentables

Theorem. For an accessible $F\colon \mathscr{C} \to \mathscr{C}$ on an accessible category $\mathscr{C}$ having binary coproducts of presentable objects,

initial F-algebra = colimit of all recursive F-coalgebras with presentable ('finite') carrier

☺ Instances: λ-presentable, finite

Recursive F-Coalgebra (R, r)



$$\begin{array}{ccc} FR & \xrightarrow{Fh} & FA \\ r \uparrow & & \downarrow \forall\, a \\ R & \xrightarrow{\exists! h} & A \end{array}$$

☺ well-founded induction

Osius '74

☺ divide & conquer algorithms (e.g. quicksort)

Capretta, Uustalu, Vene '06

Finiteness in a category $\mathscr{C}$

Fix a class Fil of filtered diagrams

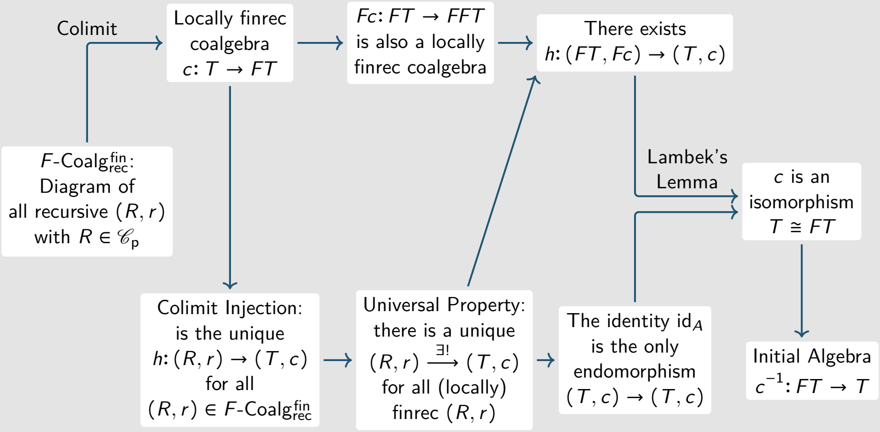
~~require cardinals~~

X (Fil-)presentable :⟺ $\mathscr{C}(X, -)$ preserves colimits of diagrams in Fil

finite

$\mathscr{C}$ (Fil-)accessible :⟺ build all objects from a set $\mathscr{C}_p$ of presentables

Proof Structure



Agda Formalization

`arxiv.org/src/2405.09504/anc/index.html`

>5000 lines (29 files)

using `agda-categories` 0.2.0

Agda Formalization

`arxiv.org/src/2405.09504/anc/index.html`

>5000 lines (29 files)

using `agda-categories` 0.2.0

? Why does the colimit of all presentable (‘finite’) recursive coalgebras exist? ?

Agda Formalization

`arxiv.org/src/2405.09504/anc/index.html`

>5000 lines (29 files)

using `agda-categories` 0.2.0

? Why does the colimit of all presentable ('finite') recursive coalgebras exist? ?

Set Theory

presentable/finite coalgebras:
(up to iso) just a set

finite recursive coalgebras:
a smaller set

↑ less elements

Agda Formalization

`arxiv.org/src/2405.09504/anc/index.html`

>5000 lines (29 files)

using `agda-categories` 0.2.0

? Why does the colimit of all presentable ('finite') recursive coalgebras exist? ?

Set Theory

presentable/finite coalgebras:
(up to iso) just a set

finite recursive coalgebras:
a smaller set

↑ less elements

Agda's Type Theory

finite coalgebras:
on the set level ℓ

finite recursive coalgebras:
one level higher $\ell + 1$
(*for all* algebras . . .)

↑ more properties

Agda Formalization

`arxiv.org/src/2405.09504/anc/index.html`

>5000 lines (29 files)

using `agda-categories` 0.2.0

? Why does the colimit of all presentable ('finite') recursive coalgebras exist? ?

Set Theory

presentable/finite coalgebras:
(up to iso) just a set

finite recursive coalgebras:
a smaller set

less elements

Agda's Type Theory

finite coalgebras:
on the set level ℓ

finite recursive coalgebras:
one level higher $\ell + 1$
(*for all* algebras . . .)

LEM to the rescue

a finite coalgebra is
recursive or not recursive

Agda: Choice of Representatives

Given $x \in C$ for a colimit $C = \mathrm{colim}(\mathscr{D} \xrightarrow{D} \mathsf{Set})$, ...

can we choose $i \in \mathscr{D}$ and $x_0 \in Di$ such that $x = \mathsf{inj}_i(x_0)$?

injection
$\mathsf{inj}_i \colon Di \to C$

Agda: Choice of Representatives

Given $x \in C$ for a colimit $C = \mathrm{colim}(\mathscr{D} \xrightarrow{D} \mathrm{Set})$, ...

can we choose $i \in \mathscr{D}$ and $x_0 \in Di$ such that $x = \mathrm{inj}_i(x_0)$?

injection $\mathrm{inj}_i \colon Di \to C$

In agda-categories ...

YES, because categories are enriched over

$$\text{Setoids} = \text{sets} + \text{equivalence relation} \approx$$

and forming quotients (e.g. in colimits) just makes $\approx$ coarser.

Open Question:

What if proper type quotients are used instead of Setoids?
(e.g. cubical agda ...)

Three Fixed Points

Three fixed points of F characterized and constructed by colimits:

F-algebra	F-coalgebra	colimit of …
initial	terminal recursive	all recursive F-coalgebras with presentable ('finite') carrier
initial iterative	terminal locally fp	all F-coalgebras with presentable ('finite') carrier
initial *cia*	terminal	all F-coalgebras

rational fixed point → (initial iterative)

completely iterative algebra → (*cia*)

In Agda

Theorem. For $F\colon \mathscr{C} \to \mathscr{C}$ accessible on an accessible category $\mathscr{C}$ having binary coproducts of presentable objects,

initial F-Algebra = colimit of all recursive F-coalgebras with presentable carrier

☺ Instances: λ-presentable, finite

Paper

Agda Doc

In Agda

Theorem. For $F\colon \mathscr{C} \to \mathscr{C}$ accessible on an accessible category $\mathscr{C}$ having binary coproducts of presentable objects,

initial F-Algebra = colimit of all recursive F-coalgebras with presentable carrier

☺ Instances: λ-presentable, finite

Conclusions & Future Work

- Agda formalization is challenging
- Decision procedure for recursiveness of 'finite' coalgebra? (its type: $\forall (C, c)\colon C$ finite $\to$ recursive$(c) \vee \neg$recursive(c))
- Similar theorem for well-founded coalgebras?
- Concrete example for a non-finitary functor in Agda

↑ long chain!

Paper

Agda Doc

- Adámek, Jiří. "Free algebras and automata realizations in the language of categories". eng. *Commentationes Mathematicae Universitatis Carolinae* 015.4 (1974), pp. 589–602. URL: http://eudml.org/doc/16649.
- Capretta, Venanzio, Tarmo Uustalu, Varmo Vene. "Recursive coalgebras from comonads". *Inf. Comput.* 204.4 (2006), pp. 437–468. DOI: 10.1016/j.ic.2005.08.005. URL: https://doi.org/10.1016/j.ic.2005.08.005.
- Osius, Gerhard. "Categorical set theory: A characterization of the category of sets". *Journal of Pure and Applied Algebra* 4.1 (1974), pp. 79–119. ISSN: 0022-4049. DOI: https://doi.org/10.1016/0022-4049(74)90032-2. URL: https://www.sciencedirect.com/science/article/pii/0022404974900322.

Initial Algebras

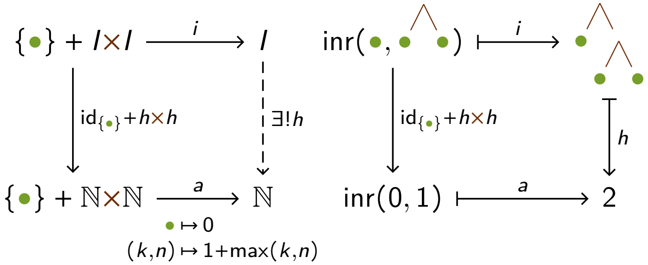
For $F\colon \mathscr{C} \to \mathscr{C}$

F-Algebra: $A \in \mathscr{C}$ with $a\colon FA \to A$.

Initial F-Algebra: unique homomorphism to every F-algebra.

Initial Algebra for $FX = \{\bullet\} + X{\times}X$

Initial F-algebra is carried by I = *all* binary trees



For $F\colon \mathscr{C} \to \mathscr{C}$

F-Coalgebra: $C \in \mathscr{C}$ with $c\colon C \to FC$.
Coalgebra-to-Algebra morphism: $s\colon (D, d) \to (A, a)$

$$\begin{array}{ccccccc}
C & \xrightarrow{g} & D & \xrightarrow{s} & A & \xrightarrow{h} & B \\
{\scriptstyle c}\downarrow & & {\scriptstyle d}\downarrow & & \uparrow{\scriptstyle a} & & \uparrow{\scriptstyle b} \\
FC & \xrightarrow{Fg} & FD & \xrightarrow{Fs} & FA & \xrightarrow{Fh} & FB
\end{array}$$

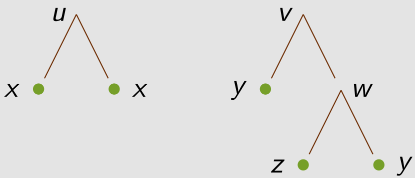
Coalgebra (D, d) is *recursive* if

for all $a\colon FA \to A$, there is a unique
Coalgebra-to-Algebra morphism $s\colon (D, d) \to (A, a)$.

Under mild conditions ...

Recursive = Well-Founded = ‘no infinite path’

R	r	FR
u	$\mapsto$	$\mathrm{inr}(x, x)$
v	$\mapsto$	$\mathrm{inr}(y, w)$
w	$\mapsto$	$\mathrm{inr}(z, y)$
x	$\mapsto$	$\mathrm{inl}(\bullet)$
y	$\mapsto$	$\mathrm{inl}(\bullet)$
z	$\mapsto$	$\mathrm{inl}(\bullet)$



For every $b\colon FB \to B$:

$$h\colon R \longrightarrow B$$

$$h(x) := h(y) := h(z) := b(\mathrm{inl}(\bullet))$$

$$h(u) := b(\mathrm{inr}(h(x), h(x)))$$

$$h(w) := b(\mathrm{inr}(h(z), h(y)))$$

$$h(v) := b(\mathrm{inr}(h(y), h(w)))$$

Definition: A category $\mathscr{D}$ is filtered if ...

- $\mathscr{D}$ is non-empty,
- for every $X, Y \in \mathscr{D}$, there is an *upper bound* $Z \in \mathscr{D}$, that is, there are morphisms $X \to Z$ and $Y \to Z$,
- for every $f, g\colon X \to Y$, there is some $Z \in \mathscr{D}$ and some $h\colon Y \to Z$ such that $h \circ g = h \circ f$.

Finitary functor = functor preserving colimits of filtered diagrams

Definition: An object $X \in \mathscr{C}$ is finitely presentable if ...

$\mathscr{C}(X, -)\colon \mathscr{C} \to \mathsf{Set}$ preserves filtered colimits.

Examples

Sets/Graphs/Posets: finite sets/graphs/posets
Nom: orbit-finite nominal sets
Vector-spaces: finite dimensional vector spaces.
Monoids: defined by finitely many generators + equations.

Category S/X: for a set $S \subseteq \mathbf{obj}\,\mathscr{C}$ and $X \in \mathscr{C}$

- objects (S, f) for $S \in \mathscr{S}$ and $f\colon S \to X$ (in $\mathscr{C}$), and
- morphisms $h\colon (S, f) \to (T, g)$ for $h\colon S \to T$ with $g \circ h = f$

Canonical Diagram

$$U_{S/X}\colon S/X \to \mathscr{C} \qquad (P \to X) \mapsto P$$

Category $\mathscr{C}$ is locally finitely presentable, if ...

it is cocomplete and
has a set S of finitely presentable objects such that
$X = \operatorname{colim} U_{S/X}$ for all $X \in \mathscr{C}$.

Fix a class Fil of small filtered categories.
(realized as a predicate in Agda)

Definition

1. An object $X \in \mathscr{C}$ is (Fil-)presentable if its hom functor $\mathscr{C}(X, -)$ preserves colimits of diagrams $D\colon \mathscr{D} \to \mathscr{C}$ with $\mathscr{D} \in \mathsf{Fil}$.
2. A category $\mathscr{C}$ is Fil-accessible provided that
   - there is a set $\mathscr{C}_\mathsf{p} \subseteq \mathscr{C}$ of (Fil-)presentable objects,
   - for all $X \in \mathscr{C}$, the coslice category $(\mathscr{C}_\mathsf{p}/X)$ lies in Fil,
   - for all $X \in \mathscr{C}$, the object X is the colimit of

$$U_{\mathscr{C}_\mathsf{p}/X}\colon \mathscr{C}_\mathsf{p}/X \to \mathscr{C} \qquad (P \to X) \mapsto P.$$